# Oefeningen Xml Technologie Xsl

Vertrek voor deze oefeningen altijd van de file *sjabloon 1.xsl*. Geef als source document *bars.xml* op, ook als je er niets uit leest. Voor EditIx is dit altijd noodzakelijk, anders wordt er niets gerund.

1. Schrijf een xsl script dat de volgende xml file genereert (zonder te lezen uit *bars.xml*):

<?xml version="1.0" encoding="UTF-8"?>

<values>

<value>45</value>

<value>55</value>

<value>65</value>

</values>

Gebruik hiervoor enkel <xsl:element>.

1. Schrijf een xsl script dat de volgende xml file genereert (zonder te lezen uit *bars.xml*):

<?xml version="1.0" encoding="UTF-8"?>

<values>

<value number="1">45</value>

<value number="2">55</value>

<value number="3">65</value>

</values>

Gebruik hiervoor <xsl:element> en <xsl:attribute>.

1. Schrijf een xsl script dat je toepast op de input file *bars.xml*. Loop door alle bar-elementen met <xsl:for-each> en genereer de volgende xml file :

<?xml version="1.0" encoding="UTF-8"?>

<metingen>

<meting number="1">90</meting>

<meting number="2">40</meting>

<meting number="3">130</meting>

<meting number="4">140</meting>

<meting number="5">120</meting>

<meting number="6">60</meting>

<meting number="7">70</meting>

<meting number="8">80</meting>

<meting number="9">120</meting>

<meting number="10">100</meting>

<meting number="11">25</meting>

<meting number="12">20</meting>

</metingen>

In het *select* attribuut van de for-each gebruik je XPath, en in de <xsl:value-of> gebruik je de punt operator. Ook de function position() moet je gebruiken.

1. Schrijf een xsl script dat je toepast op de input file *bars.xml*. Loop door alle bar-elementen en genereer de volgende xml file :

<?xml version="1.0" encoding="UTF-8"?>

<metingen>

<meting>130</meting>

<meting>140</meting>

<meting>120</meting>

<meting>120</meting>

</metingen>

Dat zijn dus maar 4 waarden. Zet daarvoor in je for-each loop de juiste XPath conditie. In woorden: alle values die groter of gelijk zijn dan 120 EN kleiner of gelijk aan 140.

Een voorbeeld:

<http://www.w3schools.com/xsl/xsl_for_each.asp> .

Check page save in examen folder

Surf voor de logische operatoren eens naar:

<http://msdn.microsoft.com/en-us/library/ms256081.aspx> .

Check page save in examen folder

1. Schrijf een xsl script dat je toepast op de input file *bars.xml*. Loop door alle bar-elementen en genereer de volgende xml file :

<?xml version="1.0" encoding="UTF-8"?>

<metingen>

<meting>90</meting>

<meting>100</meting>

</metingen>

Dat zijn dus enkel de values die groter zijn dan 80 en kleiner dan 120. Zet daarvoor in je for-each loop de juiste XPath conditie.

1. Schrijf een xsl script dat je toepast op de input file *bars.xml*. Loop door alle bar-elementen en genereer de volgende xml file :

<?xml version="1.0" encoding="UTF-8"?>

<metingen>

<meting number="1">130</meting>

<meting number="2">140</meting>

<meting number="3">120</meting>

<meting number="4">120</meting>

</metingen>

Enkel de waarden groter dan 100. Om de waarden van de attributen te vinden gebruik je de function position().

1. Schrijf een xsl script dat je toepast op de input file *bars.xml*. Genereer de volgende xml file:

<?xml version="1.0" encoding="UTF-8"?>

<metingen>

<sum>995</sum>

</metingen>

Gebruik de functie ‘sum’ met het juiste XPath argument.

1. Schrijf een xsl script dat je toepast op de input file *bars.xml*. Genereer de volgende xml file :

<?xml version="1.0" encoding="UTF-8"?>

<metingen>

<minimum>20</minimum>

<maximum>140</maximum>

<average>82.91666666666667</average>

</metingen>

Let niet op de cijfers na de comma. Gebruik de functies ‘min’, ‘max’ en ‘avg’ met het juiste XPath argument.

1. Schrijf een xsl script dat je toepast op de input file *bars.xml*. Genereer de volgende xml file :

<?xml version="1.0" encoding="UTF-8"?>

<metingen>

<meting number="1">-7.083333333333329</meting>

<meting number="2">42.91666666666667</meting >

<meting number="3">-47.08333333333333</meting >

<meting number="4">-57.08333333333333</meting >

<meting number="5">-37.08333333333333</meting >

<meting number="6">22.91666666666667</meting >

<meting number="7">12.916666666666671</meting >

<meting number="8">2.9166666666666714</meting >

<meting number="9">-37.08333333333333</meting >

<meting number="10">-17.08333333333333</meting >

<meting number="11">57.91666666666667</meting >

<meting number="12">62.91666666666667</meting >

</metingen>

Let niet op de cijfers na de comma. Vergeet de attributen niet. Elke waarde = average – value.

1. Schrijf een xsl script dat je toepast op de input file *bars.xml*. Genereer de volgende xml file :

<?xml version="1.0" encoding="UTF-8"?>

<metingen>

<meting number="1">0.09045226130653267</meting>

<meting number="2">0.04020100502512563</meting>

<meting number="3">0.1306532663316583</meting>

<meting number="4">0.1407035175879397</meting>

<meting number="5">0.12060301507537688</meting>

<meting number="6">0.06030150753768844</meting>

<meting number="7">0.07035175879396985</meting>

<meting number="8">0.08040201005025126</meting>

<meting number="9">0.12060301507537688</meting>

<meting number="10">0.10050251256281408</meting>

<meting number="11">0.02512562814070352</meting>

<meting number="12">0.020100502512562814</meting>

</metingen>

Elke waarde = value gedeeld door de som van de values. De som van deze getallen is 1, dus dit zijn eigenlijk percentages.